

# Sustainable Paths for Leadership Computing

Rick Stevens

Argonne National Laboratory

The University of Chicago

# What is Possible Over the Next 10 years

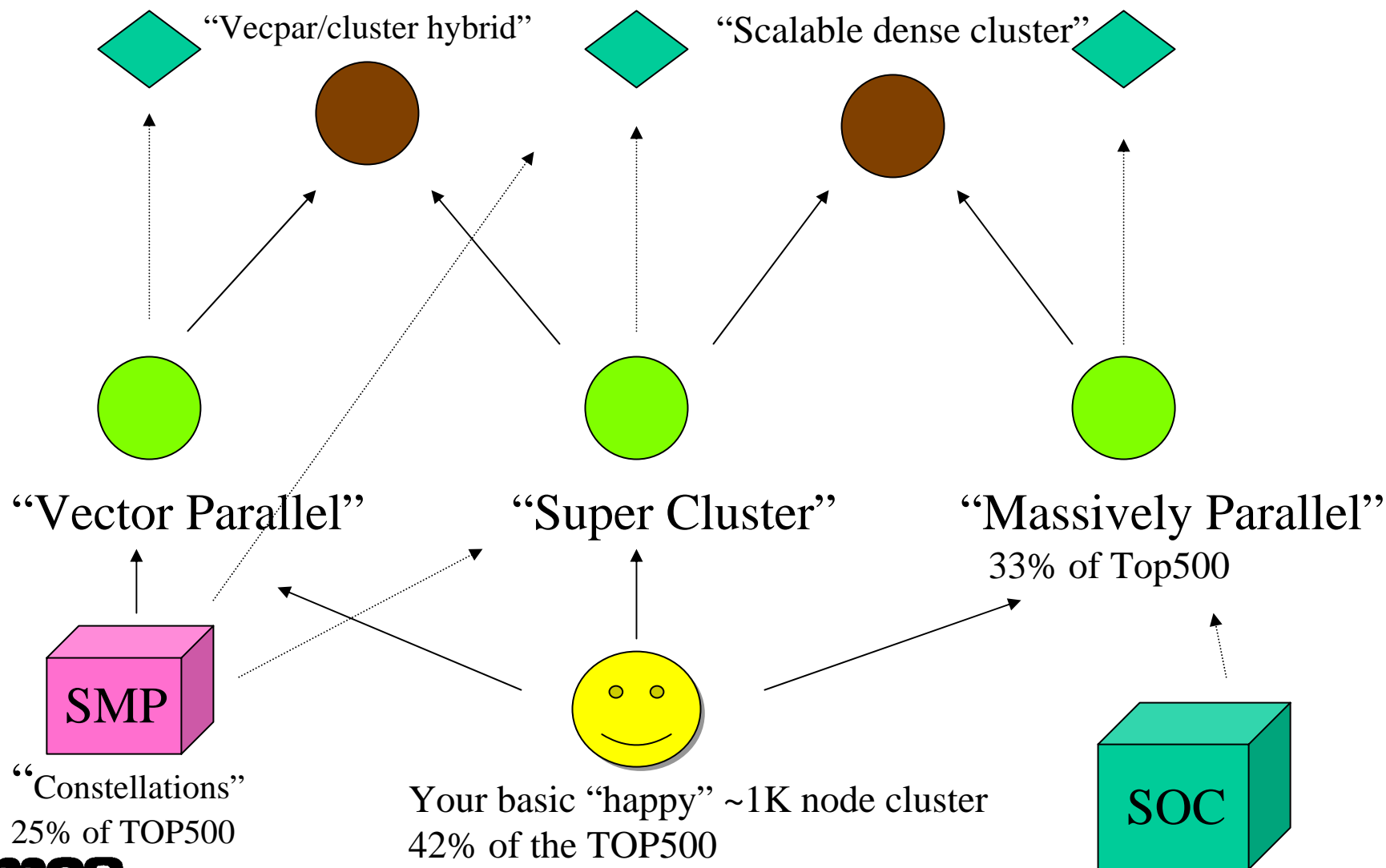
---

- Systems that can sustain a Petaflops/sec by 2010 and perhaps Exaflops by 2020
- Trends/Constraints/Requirements
  - Concurrency will be required to increase from order  $10^5$ - $10^6$  today to  $10^9$ - $10^{10}$  (ops x threads x processes x etc)
  - Power will need to be ~1-10 MW per system
  - Footprint < 40,000 sq ft, much better at <10,000 sq ft
  - Cost will need to be held constant at ~\$100-200M per system
  - Systems software will need to be 90% leveraged from open source and common with general HPC systems
  - Can run existing codes with relatively small amount of re-engineering
  - Have significant family resemblance from generation to generation
  - Economically viable to build, sell and support

# A Simplified Roadmap to Architectural Paths

three paths were outlined in the 1994 purple petaflops book

---



# A Model for Sustainability

---

1. A set of well defined science and engineering problems “Grand Challenges”  $\Rightarrow$  mission drivers (health, energy, national security, etc.)
2. Communities of critical mass  $\Rightarrow$  disciplines with active and growing populations, graduate programs, academic departments etc.
3. Robust body (ecosystem) of software  $\Rightarrow$  community codes, open source infrastructures, tools, etc.
4. Architectures (& programming models) that can host applications systems software over multiple generations of codes and users  $\Rightarrow$  preservation of investment while enabling exponential increases in performance at constant cost.

# The Blue Gene Consortium

---

- Goals
  - Provide new capabilities to selected applications partnerships
  - Provide functional requirements for Petaflops/sec version of BG
  - Build a community around a new class of HPC architecture
    - 30 university and lab partners
    - ~10 HW partners + ~20 SW collaborators
  - Develop a new (sustainable) model of partnership
    - “research product” by passing normal “productization” process/costs
    - Community based support model (hub and spoke)
  - (re-)Engage computer science researchers with HPC architecture
    - Broad community access to hardware systems
    - Enable scalable OS research and novel software research
  - DOE, NSF, NIH, NNSA, IBM partnership
    - CS research, computational science, architecture development
  - Kickoff meeting is 27th April, 2004 in Chicago

# BG/ and Possible Paths to Petaflops

---

- Potential successor machines to BG/L maybe capable of reaching petaflops/sec performance on some applications.
- One possible goal of the BG Consortium could be to help foster interest in a follow on project to BG/L to build a petascale class system (BG/P)
- One goal of the consortium could be to provide sustained (apps and ss) input during the design and development process to improve BG/P
- Another goal might be to develop an applications community able to exploit BG/X class architectures

# Community Evaluation of BG/L

---

- Diverse set of users to understand and to evaluate BG/L for important applications
  - Aim is to get 30-50 applications up on BG/L
  - Develop performance and scaling models for each
- Evaluation of:
  - Hardware (CPU/network structure)
  - Programming model (with limitations)
  - Usage model (space shared, I/O structure etc.)
  - Scalability of the machine (balance)

# Architectures and Programming Models

---

- Creation and adoption of new programming models lag significantly availability of new architectures
- The opportunity cost is high (perhaps too high) for architectures that can't leverage existing (and broadly deployed) programming models

⇒

- The price/performance advantage for a new (or re-emerging) architecture must be extreme to overcome the barrier to adopting (or re-adopting) a different programming model
- How extreme is extreme ?
  - In the near term an advantage of at least 10:1 is probably needed
  - For radical programming model changes like that needed for FPGAs the ratio is more like 100:1
  - This ratio needs to be maintained for several generations of hardware

# Conclusions

---

- Sustainability is a function of the “HPC ecosystem” as a whole not a function of the individual elements
- For the next 10 years and perhaps beyond the US should pursue a path of multiple architectures for leadership computing
  - Balancing diversity, risk and development capital
  - Current level of diversity appears adequate
- A close intellectual coupling of architecture and applications is warranted
- Awareness of the overall costs/impact of the scientific computing enterprise is required

# Provide feedback on systems related to BG/L

---

- Detailed feedback on what works and what doesn't
- Functional requests based on extensive usage on BG/L
- Performance of the networks
- Performance of the memory/caches
- Floating point performance
- Novel use of the second CPU
- Software architecture feedback
- Usage model feedback
- I/O architecture feedback
- Etc.

# A Set of Well Defined Long-Term Problems

---

- Ideal problems need to be “deep” in that the more you know the more you need to know  $\Rightarrow$  expanding a field
- Examples: understanding a cell, modeling a supernovae, understanding the brain, designing nanodevices, predicting the market
- Counter-Examples: playing chess, airline reservations systems\*, bridge design, crash simulation\*, circuit design\*
- Key Point: Need problems that can “drive” many generations of hardware and software

# Critical Mass of the Community

---

- Need areas that are already large or have significant future growth potential
- Examples: bioinformatics, ME/CFD, chem/nano, environment/climate modeling, healthcare, neuroscience, social agent models
- Counter-Examples: civil engineering, nuclear reactor engineering, nuclear weapons design, artificial intelligence
- The major drivers probably need to represent 5%-10% of the community each (e.g. 10-20 communities)

# A Robust Body of Software

---

- Each community should have multiple codes and tools to choose from, representing a diversity of algorithms, methods and technologies
- Ecosystem — rich enough to support experimentation and multiple approaches
- Leverage — can exploit libraries and tools that support more than one discipline (e.g. linear algebra, optimization)
- Opportunity — many open problems, many competitive approaches, many targets
- For each generation of hardware only 10%-20% of the software is changed